A spiking neural network model for obstacle avoidance in simulated prosthetic vision

NeuCube for obstacle avoidance in simulated prosthetic vision

Chenjie Ge, Nikola Kasabov, *Fellow, IEEE,* Jie Yang

***Abstract*—Limited by visual percepts elicited by existing visual prothesis, it's necessary to enhance its functionality to fulfill some challenging tasks for the blind such as obstacle avoidance. This paper provides a new methodology for obstacle avoidance in simulated prosthetic vision by modelling and classifying spatio-temporal (ST) video data. The proposed methodology is based on a novel spiking neural network architecture, called NeuCube as a general framework for video data modelling in simulated prosthetic vision. As an integrated environment including spiking trains encoding, input variable mapping, unsupervised reservoir training and supervised classifier training, the NeuCube consists of a spiking neural network reservoir (SNNr) and a dynamic evolving spiking neural network classifier (deSNN). First input is captured by visual prothesis, then ST feature extraction is utilized in the low-resolution prosthetic vision generated by the prothesis. Finally such ST features are fed to the NeuCube to output classification result of obstacle analysis. Experiments on collected video data and comparison with other computational intelligence methods indicate promising results. The proposed NeuCube-based obstacle avoidance methodology provides useful guidance to the blind, thus improving the current prothesis and hopefully benefiting the future prothesis wearers.**

***Index Terms*—visual prothesis, obstacle avoidance, simulated prosthetic vision, spiking neural network, NeuCube.**

I. Introduction

DEGENERATIONS of photoreceptor cells such as retinitis pigmentosa and age-related macular degeneration are devastating causes of vision loss. To restore vision to the blind, implantation of protheses may become a treatment option in the encouraging neuroengineering field. Protheses first transmit image data to an information processing unit. After electrode array gets stimulation patterns, surviving neural cells in the visual pathway can be electrically activated and visual perception can be stably evolved [1], [2], [3]. Such electrically induced visual sensations are called "phosphenes", conveying limited but useful visual information to the blind. Discernable phosphenes are usually generated in the following three locations: the visual cortex [4], the optic nerve [5], and the retina [6].

Many technical factors such as implant packaging, electrode manufactory and biocompatibility limit the maximum number of implantable electrodes, leading to a low resolution visual perception with poor understanding. Faced with such a low sampling resolution resulting in a rigorous constraint to the information expressed by pixelized images, researchers find it necessary to optimize image content in order to assist the prothesis wearers to perform better in visual tasks. More and more researches are concentrated on the how the implant recipient interprets visual information from electrical stimulation by simulation of prosthetic vision. In [7], the number of individual Chinese characters needed for accurate recognition by blind Chinese subjects is explored. Zhao et al [8] found out that distortion, dropout percentage, and pixel size have impact on the recognition of Chinese characters. At the same time, many image processing strategies are proposed in simulated prosthetic vision. Parikh et al [9] first applied saliency-based method and provided cues for region of interest detection in simulated vision. By exploring different face detection methods, Wang et al [10] concluded that such image processing methods can highlight useful information thus improving visual perception of prothesis wearers. Han et al [11] utilized feature extraction and image enhancement strategy to improve the accuracy and efficiency of object recognition. Aimed at highlighting the main object of a normal image, two different ways of pixelization [12] proved to be beneficial in daily object recognition tasks.

According to [13], there are already numerous navigation systems and tools for visually impaired individuals, among which white cane and dog guides are the most popular ones. In order to offer enough information such as speed, volume and distances to the visually impaired, a category of certain devices called electronic travel aids have been created. With the combination of different sensors such as sonars, laser scanners and cameras, information is gathered in multiple ways to guarantee the control of locomotion during navigation. However, the complexity of such systems can not ensure their reliability, robustness and overall performance, and some of them require extensive training of users to learn from different patterns output by the systems.

This study focuses on realizing the goal of obstacle avoidance for the blind based on existing visual protheses. Unlike previous studies [9], [10], [11], [12] which tried to optimize the content of phosphenes in order to improve the performances of visual tasks for the blind, our NeuCube-based obstacle avoidance system directly tells the blind the classification result of obstacle analysis, without any interaction between the prothesis wearers and the systems. As an extension of existing visual prothesis, the proposed obstacle avoidance method can make use of the down-sampled signal from information processing unit of visual prothesis, providing useful guiding information to the blind. Besides, our system is much simpler but more stable than the current navigation systems free of different sensors. To the best of our knowledge, we are the first to enhance the functionality of visual prothesis by adding additional analysis on obstacle avoidance. Relatively high accuracy is obtained owing to the use of spiking neural network based NeuCube architecture [14], [15] inspired by how human brain processes ST data. As a result, it's a natural

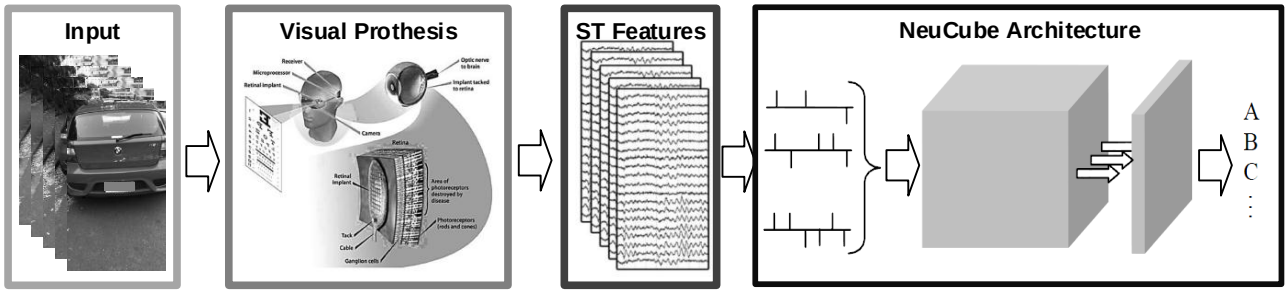

Fig. 1. The framework of our proposed NeuCube-based obstacle avoidance system.

idea to fulfill the brain-related obstacle avoidance task with the help of NeuCube, an evolving ST data processing machine.

II. The Proposed Methodology

As shown in Fig.1, first video input data is captured by visual prothesis, then we get the down-sampled signal from the output of prothesis. After that feature extraction strategy is used to obtain ST features. Finally such features are fed into NeuCube, which consists of data encoding module, NeuCube initialization module, unsupervised reservoir training module and supervised classifier training module, to output classification result of obstacle analysis. Details of the proposed model are given below.

A. ST Feature extraction

Our algorithm is based on the signal from visual prothesis, thus deemed as an extension of existing prothesis. Fig.2(a) exhibits two kinds of input video data to be classified, the one containing no obstacle in the left part and the other one containing an obstacle in the right part. Fig.2(b) shows the phosphene generated by the visual prothesis, here we use a strategy called directly lowering resolution with gaussian dots (DLR) [16] to simulate such phosphene. In each gaussian dot, central point's gray value is taken as the mean value of each grid, and the gray value of the other points around is determined by both the gray value of central point and its distance to the central point. Here only the central point of each grid is used for later processing.

As is known to us, feature extraction is important in any recognition algorithm. It's a challenging task to obtain discriminative features based on such low-resolution phosphenes. We find out that during normal walking without interference from any obstacles, only the background changes gradually in the sight. However, if one walks towards an obstacle, the size of that obstacle will become larger and larger. Based on this observation, we extract the global mean gray value of the down-sampled phosphene from each frame as ST feature. As shown in Fig.2(c), the ST feature of "no obstacle" class over time has an irregular pattern because of the changing of background, while the ST feature of "obstacle" class over time has a pattern of a relatively monotonous trend. Such two distinguishable patterns can be utilized for classification.

The shaky camera has always been an unavoidable problem in first-person videos. As for the prothesis, the same problem exists during the walking of prothesis wears. There are

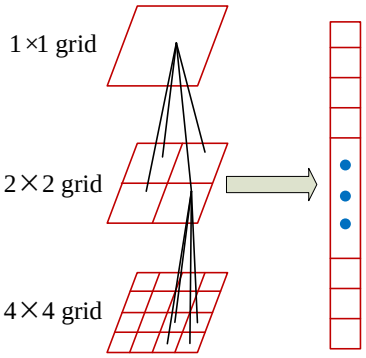

Fig. 3. Illustration of multi-scale feature extraction. Features are computed based on different partitions of input image on different layers.

many complicated methods dealing with video stabilization to remove camera motion. Here we just use a simple filtering method to alleviate the shaky effect to a certain extend. The image content has abrupt and periodic changes during the process of walking, so an averaging filter among nearby frames is used to smooth the extracted feature signal. Each feature in a certain frame is smoothed by those from other nearby frames. What's more, the gray value trends of approaching dark and bright obstacles are totally different. When one approaches a dark obstacle, the global mean value of each frame has an decreasing trend since the obstacle gradually fills his sight. As for bright obstacles, an increasing trend appears. To unite the two different occasions, we make use of an normalization step as follows where $f(t)$ and $F(t)$ denote the ST feature before and after normalization, ST features after smoothing and normalizing are shown in Fig.2(d).

$$F(t) = |f(t) - f(1)| \tag{1}$$

To fit in with obstacles of different sizes, an multi-scale strategy is considered as shown in Fig.3. On the first layer, the global gray value of the whole image is extracted. On the second layer, the entire image is divided into 4 parts and then average pooling of each grid is used to compute the average gray value of each grid. On the last layer, 16 parts are used to extract such features. Finally in a single frame, 21 ST features from different scales are extracted and stacked together.

B. data encoding

As spiking neural network can not directly process continuous signal, a data encoding method is necessary to convert the continuous signal to discrete spike trains. Threshold-based

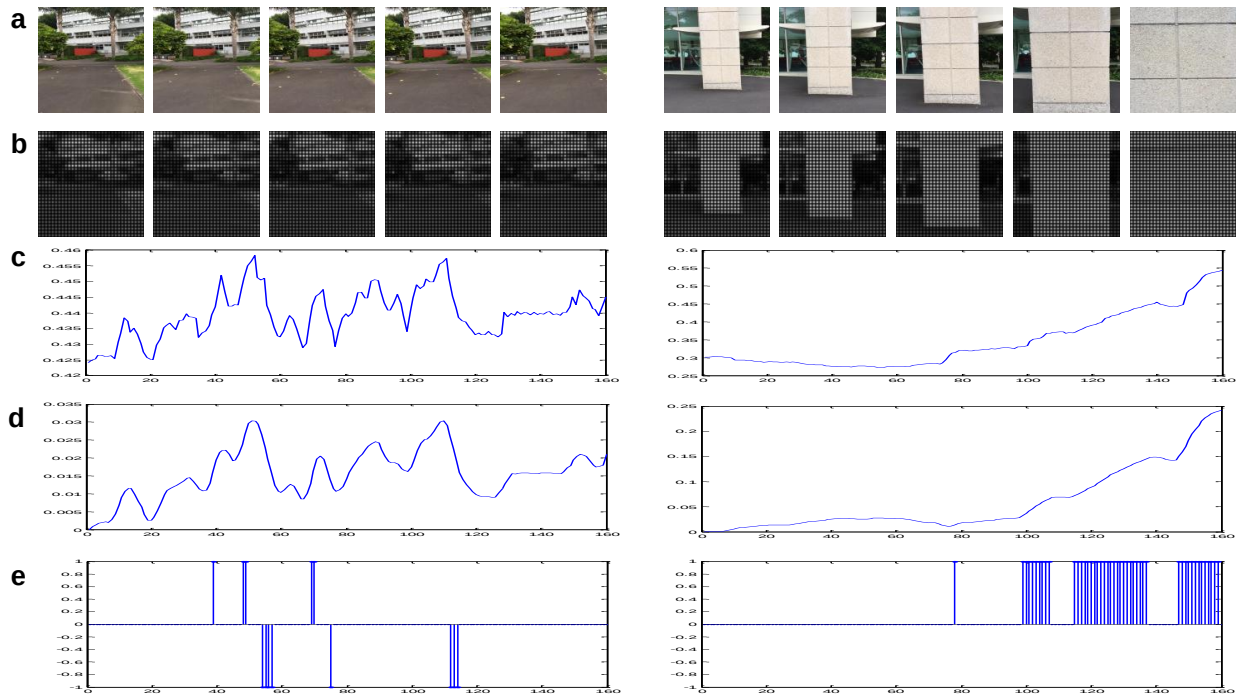

Fig. 2. Different forms of input data. (a)original input image set; (b)phosphene set; (c)original ST feature of input data; (d)ST feature after smoothing and normalizing; (e)spike representation of ST feature.

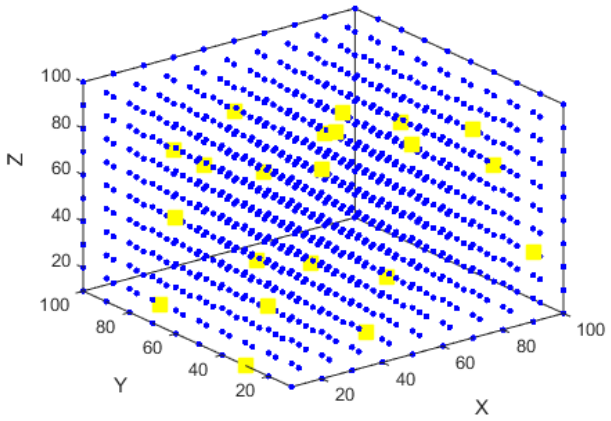

Fig. 4. A spiking neural network reservoir (SNNr) of 1000 neurons including mapped input neurons.

representation method [17] is a common method to realize such transformation, with an application for artificial retina sensor. First the gradient of original signal based on adjacent frames is computed, and then mean value m and standard deviation s of gradient signal are obtained. Threshold th is set to a combination of m and s to measure the change of original signal:

$$th = m + \alpha s \tag{2}$$

where α is a parameter controlling the tradeoff between mean value and standard deviation. Then each temporal signal is compared with its previous signal, and spikes are generated based on the gap. Specifically, a positive spike train is generated where the increasing gap is beyond the threshold, and a negative spike train is generated where the decreasing gap is beyond the threshold. We find that such encoding method is not robust to noise especially in our application where shaky camera causes meaningless fluctuations of original signal. Thus we propose a bi-direction threshold extension method to make more strict spike-generation rules. Each temporal signal is compared not only to its previous signal but also to its following signal:

$$S(t) = \begin{cases} 1, & F(t) - F(t-1) > th \ and \\ & F(t+1) - F(t) > th \\ -1, & F(t) - F(t-1) < -th \ and \\ & F(t+1) - F(t) < -th \\ 0, & otherwise \end{cases} \tag{3}$$

where $S(t)$ denotes the generated spike train. As shown in Fig.2(e), positive and negative spikes are achieved based on the changing pattern of original signal. For the "no obstacle" class, both positive and negative spikes are included because of the irregular pattern from ST feature, but the "obstacle" class contains only positive spikes although fluctuations also exist in its corresponding ST feature, which indicates that the proposed encoding method has the ability to measure the major trend of signal and alleviate the effect of noise. Discriminative spike trains are generated in this way, which is vital to the following classification based on spiking neural network.

C. NeuCube Initialization

The size of the SNNr is controlled by three parameters: n_y, n_x, n_z, which represent the number of the neurons along x, y, z directions. The size of the SNNr can vary depending on the prediction task and the data. In this experiment, a SNNr of 1000 ($10 \times 10 \times 10$) is used. SNNr is then initialized by the

Fig. 5. Examples of screenshots from the original video samples.

small world connection rule where each neuron in the reservoir is connected to its nearby neurons within a distance d, which is set to the longest distance between any pair of neurons in the reservoir multiplied by a parameter r. We initialize the weights of the connections with the product of the inverse of the distance between them and a random number within [-1,1]. Besides, 80% of the connection weights are selected to be positive and the rest 20% are negative.

For some special data such as brain EEG data [18], [19], there is always prior knowledge about the location of each collected signal channel, which can be directly used to map the signal channels into the reservoir. As for our application, there are only extracted ST features without any prior knowledge. A graph mapping method [20] is utilized to map the input variables into the reservoir based on the principle that high correlated spike trains are mapped to nearby input neurons. As shown in Fig.4, the yellow dots are the input neurons representing the different feature channels after mapping.

D. Unsupervised Reservoir Training

After mapping the ST features to SNNr, we train the SNNr with the temporal components using the Spike Timing Dependent Plasticity (STDP) learning rule [21]. It is used as an unsupervised learning phase, intending to encode hidden ST relationships from the input data into neuronal connection weights. If neuron j fires before neuron i, the connection weight from neuron j to neuron i will increase otherwise it will decrease. This ensures that the time difference in the input spiking trains, which encode the temporal patterns in the original input signals, will be captured by the neuron firing state and the unsymmetrical connection weights in the reservoir.

E. Supervised Classifier Training

The second training phase is to train the output classifier based on the association between class labels and training samples. Here the dynamic evolving Spiking Neural Networks (deSNN) [22] is used as an output classifier, owing to its ability to emphasize the importance of the first spike. For each training sample, an output neuron is dynamically created and connected to all the neurons of SNNr. Rank-order (RO) learning rule [23] is used to initialize the connection weights based on the assumption that most important information of an input pattern is contained in earlier arrived spikes, which is a phenomenon observed in biological systems as well as an important information processing concept for some spatio-temporal problems. Specifically, the connection between an postsynaptic output neuron i and any connected pre-synaptic neuron j is formulated as:

$$w_{j,i} = Mod^{order(j)} \tag{4}$$

where Mod is a modulation factor, which defines how important the order of the first spike is; $order(j)$ is the arriving order of the spikes to connection j, i among all spikes from all connection to the neuron i. Once a synaptic weight is initialized, the synapse becomes dynamic and adjusts its weight through the Spike Driven Synaptic Plasticity (SDSP) learning rule [24]. The weight increases its value with a small positive value ($Drift$) when a new spike arrives at this synapse and decreases its value with the same $Drift$ when no spike arrives.

$$\Delta w_{j,i}(t) = e_j(t) \cdot Drift \tag{5}$$

where $e_j(t) = 1$ when a spike at synapse j at time t arrives and (-1) otherwise. For every new test sample, an output neuron is created and its connection weights to other neurons of SNNr are calculated. After that, the weights are compared with those of training samples that were established during the supervised training process before. Finally the new test sample is labeled the same as an training sample whose weights are the closest to those of the test sample.

III. Experimental Design and Results

A. Design of Experiment

We collected video data by a camera with first-person view during walking, exactly simulating how the prothesis wearers walk. A $20° \times 20°$ field of view was adopted based on the available visual field of a retinal prothesis prototype [25]. 20 samples containing obstacles and 20 samples containing no obstacles were collected, and each sample was about $8s$ in duration containing 160 frames. Both indoor and outdoor scenarios were included, and the obstacles included walls, cars, trash cans, trees, statues and so forth (examples of screenshots from the original video samples are shown in Fig.5).

Here the "obstacle" class was considered as positive samples and "no obstacle" class was considered as negative samples. For quantitative comparison, we employed three evaluation metrics, true positive rate (TPR), true negative rate (TNR) and $OverallAccuracy$ to measure the classification accuracy, they are computed as follows:

$$TPR = \frac{TP}{TP + FN} \tag{6}$$

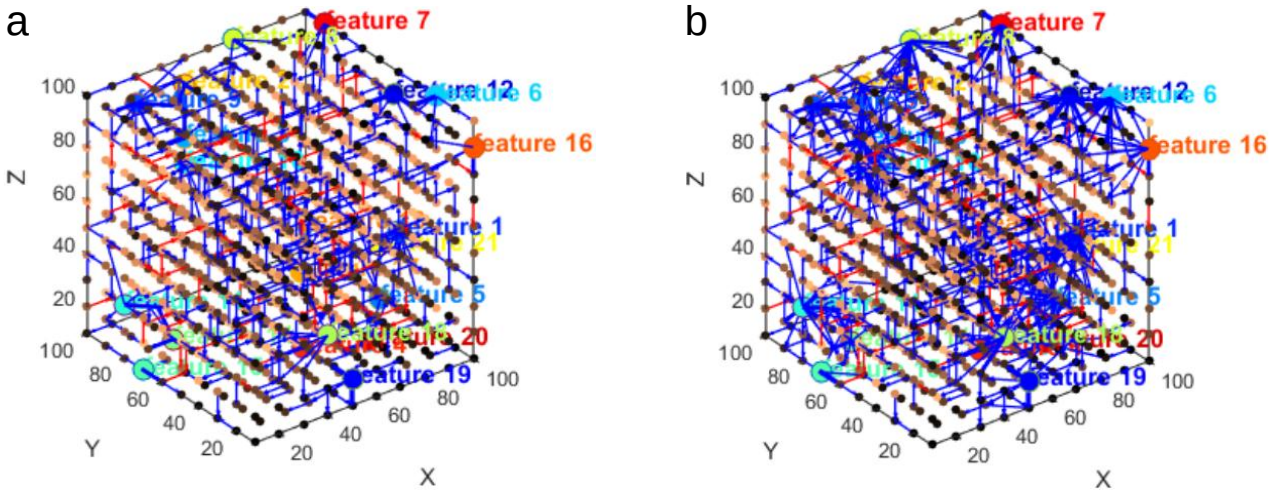

Fig. 6. (a)Connection weights in SNNr before training. (b)Connection weights in SNNr after training.

$$TNR = \frac{TN}{FP + TN} \tag{7}$$

$$OverallAccuracy = \frac{TP + TN}{TP + FP + +FN + TN} \tag{8}$$

where TP counts number of positive samples correctly classified as positive samples, FN counts number of positive samples incorrectly classified as negative samples, FP counts number of negative samples incorrectly classified as positive samples and TN counts number of negative samples correctly classified as negative samples. As a result, TPR measures the classification accuracy of positive samples, TNR measures the classification accuracy of negative samples and $OverallAccuracy$ can be deemed as a mean of TPR and TNR because the numbers of positive and negative samples are the same.

B. NeuCube Visualization

Unlike traditional computational intelligence methods for classification, the proposed NeuCube-based method has a better understanding of the data and the brain processes that generated it. As a result, NeuCube has been successfully utilized for learning, classification and comparative analysis on EEG data [18], [19] by visualizing the trained SNNr and analyzing its connectivity and spiking activity. Besides, the NeuCube visualization also shows its property of evolving, which means that any previously unknown patterns can be added to SNNr being learned and recognized without redesigning all the learning process. Such kind of property reflects the principle of brain cognitive development.

In Fig.6, the connection weights in SNNr before and after training are visualized. Blue lines mean positive connections and red lines mean negative connections. For each line, thickness represents weight value. Also, a brighter neuron in SNNr has stronger connections with other neurons than a darker neuron. Some conclusions can be derived to better understand the data and the evolving NeuCube system.

First, nearby neurons have denser connections than neurons which are far away from each other after training. This indicates that the mapping method in the NeuCube initialization step really maps the high-correlated spike trains to nearby input neurons. Because high-correlated spike trains are more time dependent with each other, it's reasonable to map them to nearby neurons in SNNr to have similar interactions with other neurons.

Second, neurons in some parts are generally more brighter than other parts, which reflects the evolving property of SNNr because the input data patterns are learned in the reservoir after training. After the unsupervised reservoir training, neurons with stronger connections to other neurons indicate more spike transmissions between neurons' synapses, thus reflecting the different importance of different features.

C. The Role of Each Step in the Proposed Methodology

To illustrate the effectiveness of the ST features, more visual examples are given in Fig.7. Fig.7(a) shows the ST feature of the "no obstacle" class, while Fig.7(b) shows the ST feature of the "obstacle" class. ST features are extracted in different samples and here only one dimension of ST features are visualized. It is observed that ST features of "no obstacle" class share irregular patterns on account of different background. However, the ST features of "obstacle" class have similar trend of increasing, indicating that they are appropriate to describe the common properties of approaching different obstacles. Therefore, the extracted ST features can help distinguish the two different classes.

We changed different steps of the proposed methodology to evaluate the role of each step through the above three evaluation metrics. Different methods varied in feature selection, feature post-processing, phosphene resolution and feature extraction strategy. The proposed method denoted as "ours" used three layers of feature, smoothing and normalizing strategy, 24×24 phosphene resolution and average pooling feature extraction. All these methods were compared with the

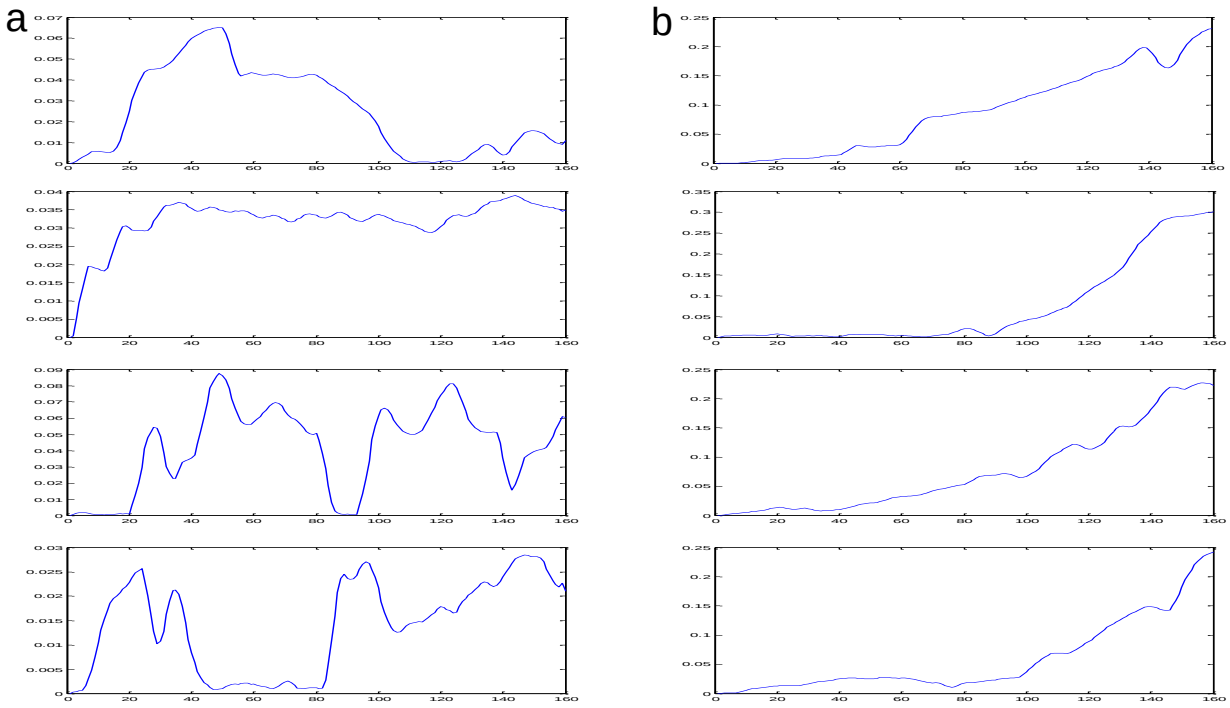

Fig. 7. (a)ST features of the "no obstacle" class. (b)ST features of the "obstacle" class.

TABLE I
ACCURACY EVALUATION WHEN CHANGING DIFFERENT STEPS OF THE PROPOSED METHODOLOGY.

Different Methods	Overall Accuracy(%)	TPR(%)	TNR(%)
1d+4d feature	82.25	76.50	88.00
16d feature	86.50	90.00	83.00
without smoothing and normalizing	75.25	68.00	82.50
24×24 phosphene resolution	88.75	87.50	90.00
max pooling	62.00	51.00	73.00
Ours	89.75	89.50	90.00

same default parameters. 10-fold cross validation was used and repeated for 10 times for more convincing and accurate results.

In the proposed methodology, three layers of features are used for further processing. We evaluated the performance of utilizing the combination of first and second layers of features, which are denoted by "1d+4d feature" in Table 1. At the same time, only the third layer of feature which is denoted by "16d feature" was used for comparison. As shown in Table 1, neither of the two methods is better than the proposed method, demonstrating the power of using three layers of features. The reason lies in that the different layer varies in different partitions of original image. The more layers used, the more different sizes of obstacles can be captured in the proposed features, thus increasing the accuracy of classification. The reason why we did not use more layers of features is that the obstacle avoidance task ought to be real-time, and the computation cost should be as low as possible. It is observed that the usage of three layers has already guaranteed a satisfying overall accuracy of nearly 89.75%.

The performance of feature post-processing steps such as smoothing and normalizing were also tested. Without smoothing or normalizing, the overall accuracy had drop to 75.25%, which indicates that the proposed smoothing and normalizing steps removed the effect of shaky camera to a certain extent and united the two different occasions of approaching dark and bright obstacles, making the two different patterns to be classified more distinguishable.

Phosphene resolution is an important factor for different tasks in prosthetic vision because the resolution determines the visual information from protheses. The visual contents received from protheses increase with phosphene resolution, which is in agreement with researches on minimal amount of visual information necessary for functional tasks such as visual acuity [26], mobility [27] and reading [28]. In our experiment, two normal phosphene resolutions 24×24 and 32×32 were tested. Their classification accuracies are quite similar in Table 1, and 32×32 resolution is a bit better. This indicates that the proposed methodology is not very sensitive to the resolution. Unlike previous studies where image processing strategy is used to enhance the simulated phosphene so that participants are required to perform certain tasks based on the phosphene,

TABLE II
ACCURACY EVALUATION USING DIFFERENT PARAMETERS.

STDP learning Rate	Mod	Drift	Overall Accuracy(%)
0.01	0.4	0.025	87.00
0.006	0.4	0.001	87.25
0.01	0.675	0.001	87.50
0.006	0.95	0.05	88.25
0.001	0.95	0.025	89.00
0.006	0.4	0.05	89.25
0.01	0.95	0.05	89.75
0.01	0.4	0.05	90.50

TABLE III
ACCURACY EVALUATION USING DIFFERENT COMPUTATIONAL INTELLIGENCE METHODS.

Classification Methods	Overall Accuracy(%)	TPR(%)	TNR(%)
Adaboost	77.50	75.00	80.00
MLP	32.50	50.00	15.00
SVM	61.25	52.50	70.00
NeuCube	90.50	88.00	93.00

the proposed methodology only extract ST features from it. Later NeuCube-based classification method outputs results automatically without any judgement from prothesis wearers. That's why the proposed methodology is not much dependent on the phosphene resolution.

Pooling is a kind of operation which combines nearby values in real space or feature space through a max or average operator [29]. The goal of pooling is to achieve a new feature representation that preserves important information while discarding irrelevant details. At the same time, the compact representations obtained lead to better robustness to noise and clutter. In the proposed methodology, average pooling is used to compute the average gray value of each grid. Here we also replaced average pooling with max pooling, where maximum gray value of each grid was extracted, to evaluate the classification accuracy. Table 1 reports that max pooling strategy achieves much lower accuracy than average pooling, which implies that noise actually exists in each feature grid and average pooling is a more stable method to extract useful feature under the circumstance of shaky camera.

D. Parameter Optimization

An important step in obtaining good results from NeuCube is the optimization of parameter values, because the output classification accuracy depends on parameter settings. In this experiment, parameter optimization was achieved via grid search, which is an exhaustive search method based on different combinations of parameters. After that, the best classification results as well as the optimal parameters could be saved and reported. Some prime parameters of NeuCube are listed as follows:

−Small world connectivity radius. Each neuron of the SNNr is initialized connected to its neighbouring neurons within this parameter as distance. In this experiment, we have used a value of 25.

−The threshold of firing, the refractory time and the potential leak rate of the LIF neurons. We use leaky integrate and fire (LIF) neuron model [30] in this experiment. When a LIF neuron receives a spike, its potential gradually increases with every input spike until it reaches an established threshold of firing. Thereafter, an output spike is emitted and the potential is reset to an initial state for a period of time which is called refractory time. Between spikes, the potential leaks at a constant rate called potential leak rate. In our experiments the three parameters are set to 0.5, 6 and 0.002 respectively.

−STDP learning rate. It's a parameter to modify the neuronal connections regarding repetitive arrived spikes to the synapses. If a neuron fires before another neuron, its connection weight increases otherwise it decreases by STDP learning rate.

−Mod: Based on the rank-order learning rule, connection weight between neuron i to neuron j is computed depending on a modulation factor Mod and the order of the first incoming spike.

−Drift: After the initial connection weights are set, the occurrence of following spikes are considered to update with respect to time. When a spike arriving from neuron i at time t after the first one was emitted, the weight increases by the value of Drift otherwise if no spike arrives it decreases.

In our experiment the performance was relatively stable when changing the small world connectivity radius, the threshold of firing, the refractory time and the potential leak rate of the LIF neurons, so the four parameters were fixed empirically. The rest three parameters including the STDP rate, Mod and Drift were optimized using an exhaustive search method. Different combinations of these parameters were tested and only the results which were closest to the best one were listed in Table 2. It is observed that the maximum accuracy reaches to 90.5%.

E. Comparison with other computational intelligence methods

To prove the validity of the proposed model, it was also compared with computational intelligence methods such as Adaboost, multiple layer perception(MLP) and support vector machine(SVM). Since we had optimized the parameters in the proposed model, the parameters for each classifier were also tuned heuristically to obtain the best results from each of them. For example, the MLP model used 20 hidden nodes, a learning rate of 0.01 and 500 iterations. The Adaboost model used 50 iterations and 1 random seed. As for the SVM classifier, quadratic kernel was chosen as the best kernel function for the data. What's more, as the spatio-temporal data was too large for these three classifiers, such data was compressed with principal component analysis method (PCA) for further classification. The result of the experiment is laid out in Table 3. The proposed NeuCube model achieved the highest overall accuracy, true positive rate and true negative rate. As a result, it is safe to say that traditional computational intelligence methods like Adaboost, MLP and SVM are not capable of learning from this type of data. The proposed NeuCube model is good at learning ST features, therefore relatively high accuracy is guaranteed.

IV. Conclusion

This paper has shown a promising and novel way to fulfill the obstacle avoidance task by modeling and classifying the ST video data in simulated prosthetic vision. It is observed that NeuCube is a powerful tool to learn the ST data and therefore can be successfully used in classifying the "obstacle" and "no obstacle" occasions for the prothesis wearers. Effective ST features are first extracted, followed by a data encoding method converting the changes of signals into spikes. Then NeuCube initializes its architecture, learns the hidden ST relationships of input spikes and trains the final classifier to output classification result of obstacle analysis. Role of each step is illustrated and new features of both NeuCube visualization and parameter optimization shows the superiority of the proposed NeuCube-based methodology. At last, more satisfying detection accuracy is obtained compared to traditional techniques like Adaboost, MLP and SVM.

Acknowledgment

This work was completed during the visit of Chenjie Ge to the Knowledge Engineering and Discovery Research Institute (KEDRI, www.kedri.aut.ac.nz). The authors would like to thank the anonymous reviewers for their helpful comments to improve this paper.

References

[1] Joseph F Rizzo, John Wyatt, John Loewenstein, Shawn Kelly, and Doug Shire, "Perceptual efficacy of electrical stimulation of human retina with a microelectrode array during short-term surgical trials," *Investigative ophthalmology & visual science*, vol. 44, no. 12, pp. 5362–5369, 2003.

[2] Eberhart Zrenner, Karl Ulrich Bartz-Schmidt, Heval Benav, Dorothea Besch, Anna Bruckmann, Veit-Peter Gabel, Florian Gekeler, Udo Greppmaier, Alex Harscher, Steffen Kibbel, et al., "Subretinal electronic chips allow blind patients to read letters and combine them to words," *Proceedings of the Royal Society of London B: Biological Sciences*, p. rspb20101747, 2010.

[3] Takashi Fujikado, Motohiro Kamei, Hirokazu Sakaguchi, Hiroyuki Kanda, Takeshi Morimoto, Yasushi Ikuno, Kentaro Nishida, Haruhiko Kishima, Tomoyuki Maruo, Kunihiko Konoma, et al., "Testing of semichronically implanted retinal prosthesis by suprachoroidal-transretinal stimulation in patients with retinitis pigmentosa," *Investigative ophthalmology & visual science*, vol. 52, no. 7, pp. 4726–4733, 2011.

[4] Wm H Dobelle, "Artificial vision for the blind by connecting a television camera to the visual cortex," *ASAIO journal*, vol. 46, no. 1, pp. 3–9, 2000.

[5] Claude Veraart, Christian Raftopoulos, J Thomas Mortimer, Jean Delbeke, Delphine Pins, Geraldine Michaux, Annick Vanlierde, Simone Parrini, and Marie-Chantal Wanet-Defalque, "Visual sensations produced by optic nerve stimulation using an implanted self-sizing spiral cuff electrode," *Brain research*, vol. 813, no. 1, pp. 181–186, 1998.

[6] Mark S Humayun, James D Weiland, Gildo Y Fujii, Robert Greenberg, Richard Williamson, Jim Little, Brian Mech, Valerie Cimmarusti, Gretchen Van Boemel, Gislin Dagnelie, et al., "Visual perception in a blind subject with a chronic microelectronic retinal prosthesis," *Vision research*, vol. 43, no. 24, pp. 2573–2581, 2003.

[7] Xinyu Chai, Wei Yu, Jia Wang, Ying Zhao, Changsi Cai, and Qiushi Ren, "Recognition of pixelized chinese characters using simulated prosthetic vision," *Artificial organs*, vol. 31, no. 3, pp. 175–182, 2007.

[8] Ying Zhao, Yanyu Lu, Chuanqing Zhou, Yao Chen, Qiushi Ren, and Xinyu Chai, "Chinese character recognition using simulated phosphene maps," *Investigative ophthalmology & visual science*, vol. 52, no. 6, pp. 3404–3412, 2011.

[9] N Parikh, L Itti, M Humayun, and J Weiland, "Performance of visually guided tasks using simulated prosthetic vision and saliency-based cues," *Journal of neural engineering*, vol. 10, no. 2, pp. 026017, 2013.

[10] Jing Wang, Xiaobei Wu, Yanyu Lu, Hao Wu, Han Kan, and Xinyu Chai, "Face recognition in simulated prosthetic vision: face detection-based image processing strategies," *Journal of neural engineering*, vol. 11, no. 4, pp. 046009, 2014.

[11] Tingting Han, Heng Li, Qing Lyu, Yajie Zeng, and Xinyu Chai, "Object recognition based on a foreground extraction method under simulated prosthetic vision," in *Bioelectronics and Bioinformatics (ISBB), 2015 International Symposium on*. IEEE, 2015, pp. 172–175.

[12] Jing Wang, Heng Li, Weizhen Fu, Yao Chen, Liming Li, Qing Lyu, Tingting Han, and Xinyu Chai, "Image processing strategies based on a visual saliency model for object recognition under simulated prosthetic vision," *Artificial organs*, 2015.

[13] Dimitrios Dakopoulos and Nikolaos G Bourbakis, "Wearable obstacle avoidance electronic travel aids for blind: a survey," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 40, no. 1, pp. 25–35, 2010.

[14] Nikola K Kasabov, "Neucube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data," *Neural Networks*, vol. 52, pp. 62–76, 2014.

[15] Nikola Kasabov, Nathan Matthew Scott, Enmei Tu, Stefan Marks, Neelava Sengupta, Elisa Capecci, Muhaini Othman, Maryam Gholami Doborjeh, Norhanifah Murli, Reggio Hartono, et al., "Evolving spatio-temporal data machines based on the neucube neuromorphic framework: Design methodology and selected applications," *Neural Networks*, 2015.

[16] Angélica Pérez Fornos, Jorg Sommerhalder, Benjamin Rappaz, Avinoam B Safran, and Marco Pelizzone, "Simulation of artificial vision, iii: do the spatial or temporal characteristics of stimulus pixelization really matter?," *Investigative ophthalmology & visual science*, vol. 46, no. 10, pp. 3906–3912, 2005.

[17] T Delbruck, "jaer open source project," 2007.

[18] Nikola Kasabov and Elisa Capecci, "Spiking neural network methodology for modelling, classification and understanding of eeg spatio-temporal data measuring cognitive processes," *Information Sciences*, vol. 294, pp. 565–575, 2015.

[19] Maryam Gholami Doborjeh, Grace Wang, Nikola Kasabov, Robert Kydd, and Bruce Roy Russell, "A spiking neural network methodology and system for learning and comparative analysis of eeg data from healthy versus addiction treated versus addiction not treated subjects," 2015.

[20] Enmei Tu, Nikola Kasabov, Marini Othman, Yuxiao Li, Susan Worner, Jie Yang, and Zhenghong Jia, "Neucube (st) for spatio-temporal data predictive modelling with a case study on ecological data," in *Neural Networks (IJCNN), 2014 International Joint Conference on*. IEEE, 2014, pp. 638–645.

[21] Sen Song, Kenneth D Miller, and Larry F Abbott, "Competitive hebbian learning through spike-timing-dependent synaptic plasticity," *Nature neuroscience*, vol. 3, no. 9, pp. 919–926, 2000.

[22] Nikola Kasabov, Kshitij Dhoble, Nuttapod Nuntalid, and Giacomo Indiveri, "Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition," *Neural Networks*, vol. 41, pp. 188–201, 2013.
[23] Simon Thorpe and Jacques Gautrais, "Rank order coding," in *Computational Neuroscience*, pp. 113–118. Springer, 1998.
[24] Stefano Fusi, Mario Annunziato, Davide Badoni, Andrea Salamon, and Daniel J Amit, "Spike-driven synaptic plasticity: theory, simulation, vlsi implementation," *Neural computation*, vol. 12, no. 10, pp. 2227–2258, 2000.
[25] David D Zhou, Jessy D Dorn, and Robert J Greenberg, "The argus® ii retinal prosthesis system: An overview," in *Multimedia and Expo Workshops (ICMEW), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–6.
[26] Kichul Cha, Kenneth Horch, and Richard A Normann, "Simulation of a phosphene-based visual field: visual acuity in a pixelized vision system," *Annals of biomedical engineering*, vol. 20, no. 4, pp. 439–449, 1992.
[27] Kichul Cha, Kenneth W Horch, and Richard A Normann, "Mobility performance with a pixelized vision system," *Vision research*, vol. 32, no. 7, pp. 1367–1372, 1992.
[28] Kichul Cha, Duane K Boman, Kenneth W Horch, and Richard A Normann, "Reading speed with a pixelized vision system," *JOSA A*, vol. 9, no. 5, pp. 673–677, 1992.
[29] Y-Lan Boureau, Jean Ponce, and Yann LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 111–118.
[30] Wulfram Gerstner, "What is different with spiking neurons?," in *Plausible neural networks for biological modelling*, pp. 23–48. Springer, 2001.

PLACE PHOTO HERE

Michael Shell Biography text here.

John Doe Biography text here.

Jane Doe Biography text here.